

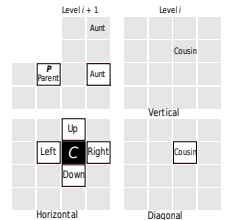
4. Übung: Lineare Nachbarschaftsfilter

Michael Grunwald

Aufgabe 1

- Aufgabe 1.1 Implementieren Sie eine MEX Funktion (C/C++), welche es erlaubt ein Bild (im sinnvollen Rahmen) mit frei wählbaren Filtermasken zu falten.
 - Prototyp: `[out_image] = mxFilter(in_image, filter, off);`

`out_image` Ergebnisbild nach Faltung von `in_image` mit `filter`
`in_image` Eingangsbild (uint8)
`filter` Filtermatrix (double)
`off` Offset (uint8)
 - 8-Bit Graustufenbilder als Eingangs- und Ausgangsdaten.
 - Filtermatrix der Größe (N×N) mit $N = (2K + 1)$, $K = 1, 2, \dots$
 - Ankerpunkt (Hot Spot) ist die Mitte der Filtermatrix.
 - Beispielaufruf: `mxFilter(imread('bild.jpg'), [1 1 1; 1 3 1; 1 1 1]./11, uint8(0));`
- Aufgabe 1.2 Laden Sie das Buch Digitale Bildverarbeitung herunter und lesen Sie Kapitel 6 (Filter). Bitte beachten Sie, dass der Link nur intern im HTWG Netz funktioniert.



Aufgabe 2

- Erweitern Sie die Funktion `mxFilter.cpp` um folgende Randbedingungen:
 - min
Setzt Bildpunkte auf den minimalen Wert (0)
 - max
Setzt Bildpunkte auf den maximalen Wert (255)
 - mirror
Spiegelt den Bildinhalt an den Bildkanten
 - continue
Setzt das Bild außerhalb mit dem gleichen Pixelwert, wie das entsprechende am nächsten liegende Randpixel, fort.
- Prototyp: `[out_image] = mxFilter(in_image, filter, off, edge);`
`edge`: Parameter zur Auswahl der Randbehandlung ('min') – String
- Untersuchen Sie die Randbehandlungen auf ihr Verhalten bei Benutzung verschiedener Filter.

4. Übung: Lineare Nachbarschaftsfilter

Michael Grunwald

Aufgabe 3

Beantworten Sie folgende Fragen (siehe Buch ab S. 99).

- Aufgabe 3.1 Nennen Sie die Arten und Eigenschaften von linearen Filtern.
- Aufgabe 3.2 Was ist der Unterschied zwischen linearen und nichtlinearen Filtern?

Bemerkung

- Bitte erstellen Sie für die verwendeten cpp-Dateien die zugehörigen kommentierten Matlab m-Dateien.
- Die Aufgaben werden elektronisch (m.grunwald@htwg-konstanz.de) und per Ausdruck abgegeben. (Ausdruck: mxFilter.cpp und Beantwortung der Aufgabe 3, Elektronisch: Alle cpp-Dateien und Aufgabe 3 als pdf-Datei)